

## AIS Programming Standards

# QMF Standards

- [Introduction](#)
- [QMF Environment](#)
- [QMF Profile](#)
- [QMF Governor](#)
- [Naming Conventions](#)
- [Custom QMF Commands](#)
- [CLISTS](#)
- [SPF Panels](#)
- [Printing Issues](#)
- [Batch Processing](#)
- [QMF Technical Support & Training Issues](#)
- [Related Policies, Standards, Processes](#)

## Introduction

The following are the standards and guidelines that should be deployed by all application teams as they work with QMF on the AIS mainframe. The guidelines listed cover both basic QMF issues as well as topics related to the creation of a customized QMF environment for AIS business partners.

## QMF Environment at AIS

The DB2Q subsystem was created to separate QMF usage and tables from the production DB2P environment. This was required to ensure that the production CICS applications running are not impacted by the ad hoc queries that are submitted by end users in QMF. The separation eliminates problems that might occur with shared usage of buffer space and system catalogue tables.

- QMFT - invokes QMF using DB2T
- QMFP - invokes QMF using DB2Q

## QMF Profile

Every user has their own 'profile' defined with default settings. The QMF technical manuals list the standard default settings. The following is a list of recommended changes that can be made to a user profile.

## General Operands

CASE value should be 'STRING'. This will upper case your input yet leave everything in quotes in whatever case was typed. This is required for data stored in upper/lower case. This makes it easy to use QMF commands which must be in upper case.

The other General Operands are personal preferences.

## Defaults for Printing

- **WIDTH** should be 150 for maximum landscape printing.

- **LENGTH** should be 60 for the number of lines on a page.
- **PRINTER** should be blank since there are no defined printers for QMF. Each application can decide how to do their own printing.

## QMF Administration Operands

SPACE identifies the DB2 database or tablespace where data is stored when the QMF SAVE command is used. This requires that the appropriate authority be given to selected users when used in the production environment.

One advantage in using the SPACE option is the creation of temporary tables. These tables are useful for simplifying queries that require a lot of joins, especially outer joins. These temp tables require the user have create authority in the tablespace used since the QMF SAVE command creates these tables. Also note there are no indexes on these tables. It is also up to the user or application team to manage the space being used. These are temp tables and should be dropped after they are no longer needed.

TRACE should be NONE.

## QMF Governor

AIS has set database resource limits on queries or procedures that you run. If your query or procedure exceeds a time limit or retrieves more rows from the database than the limit set by AIS, processing is interrupted. A panel is displayed that lets you specify whether you meant to continue or cancel the query or procedure.

The QMF Governor must be set high enough to support background QMF processing as well as foreground processing since there is only one governor for both environments.

## Naming Conventions

- The following rules apply when naming objects saved in the database.
  - Names for queries, forms, procedures, tables and views must be unique.
  - Names cannot start with a number.
  - Avoid using special character in a QMF object name, such as,  
., ; : < > 9 0 | + - \* / = & ^ ' "
  - A name cannot be longer than 18 characters. This does not include the owner name, etc.
  - Avoid using QMF reserved words in an object name.
- Recommended naming conventions include using a suffix identifier for the object saved. For example, queries should end with \_q, procs with \_p, forms with \_f, etc.
- Use the same base name (non-suffix part) for objects that go together. For example, abc\_p (proc) runs abc\_q (query) and abc\_f (form) is applied to the result set.

## Custom QMF Commands

QMF has the ability to allow an application to tailor the QMF environment (to a limited degree). Some of these areas are mentioned below. For more detailed information see the QMF technical manuals.

When a new QMF user desires custom command capabilities, modifications to the overall QMF profile table (q.profile) must be performed by a Systems programmer. Details as to which columns and the desired column value must be specified - ie PF keys and Command Synonym are examples of columns

that can be changed.

In conjunction with the overall QMF profile table, an application specific PF Key Table can be created to override selected or all PF keys on certain main QMF screens (popup windows in QMF cannot have their PF keys overridden). These overrides do not effect your ISPF PF key settings. They are independent. The PF key settings are kept in DB2 tables and are accessible via QMF global variables which point to the specified PF key table. The name of the PF Key Table must be included within the overall QMF profile table.

An application specific Command Synonym Table can be created to supplement your QMF commands. The default Synonym table will be overridden by your Synonym table. If you wish to keep any commands, copy them to your new synonym table. The name of the Command Synonym Table must be included within the overall QMF profile table.

Uses for these synonyms may include creating a simpler QMF command interface, such as saving objects or printing QMF reports. The QMF SAVE command does not make it overly simple to add/edit comments or even save an object. Remember, the QMF SAVE command restricts use of any object saved to your use only, by default. If you are writing application objects then you will want them to be shareable. Noted above there are no printers defined for QMF. You must create a command synonym to allow for a simple printing interface. A command synonym will allow you to override this default.

## **CLISTS / REXX**

CLISTS or REXX are the primary languages that can be used to create the command synonyms. QMF commands can be executed from a CLIST or REXX exec whenever QMF is active. If you decide to use REXX, you will gain additional flexibility by using RLX, a vendor product that permits the use of DB2 SQL from within the REXX exec.

## **SPF Panels**

SPF panels are an excellent way to quickly create a simple user interface for the new command synonyms that you have created. These are invoked from a REXX exec or CLIST. See Dialogue Management technical manuals for more information.

## **Printing Issues**

Basically, it is not easy to print (foreground) from QMF using the default methods. It is recommended the application create a synonym command to print a QMF report. A synonym will give you complete control of the output, i.e., landscape, portrait, printer formats, copies, etc. Without this synonym command, printing is tedious and not user friendly.

## **Batch Processing**

QMF gives you the ability to run queries or procedures in batch processing. This means you can create foreground or background JCL jobs to produce QMF reports or get DB2 data via QMF. Parms can also be passed to QMF, such as date, to help control you queries and make them more dynamic. Basically, anything you can do in QMF in the foreground, you can do in batch/background processing, as well. The QMF technical manuals describe this process in more detail.

## QMF Technical Support & Training Issues

- The client's department support staff must be prepared and trained to manage initial problem solving as well as the training of new QMF users.
  - There must be adequate support provided by each programming team to respond to complex questions from the departmental staff.
  - The programming team or AIS database administrator may be responsible for preparing a detailed data model. ER/win is the tool currently used by AIS for data modeling.
- 

### Related Policy, Procedures or Standards

see also:

- [Accountability & Exceptions](#)
  - Query Management Facility technical manuals
  - TSO Command Reference Manual
  - Dialogue Management Facility
  - RLX technical manual
- 

Back to [Programming Standards](#) home page.

---

Prepared by Andrea Korn;  
Last updated: July 29, 1996